International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Text Extraction from Complex Color Images Using Optical Character Recognition

Prachi R. Dussawar[1], Parul Bhanarkar Jha[2]

[1]Student (M. Tech) Wireless Communication & Computing, TGPCET/RTM Nagpur University, Nagpur, India

[2]Professor, Head of Department, Inform, TGPCET/ RTM Nagpur University, Nagpur, India

Abstract: ***Optical Character Recognition (OCR) is a system that provides a full alphanumeric recognition of printed or hand written characters by simply scanning the text image. OCR system interprets the printed or handwritten characters image and converts it into corresponding editable text document. The text image is divided into regions by isolating each line, then individual characters with spaces. After character extraction, the texture and topological features like corner points, features of different regions, ratio of character area and convex area of all characters of text image are calculated. Previously features of each uppercase and lowercase letter, digit, and symbols are stored as a template. Based on the texture and topological features, the system recognizes the exact character using feature matching between the extracted character and the template of all characters as a measure of similarity.***

Keywords: Character recognition; Feature Extraction; Feature Matching; Text extraction; Character extraction

1. Introduction

Optical character recognition (OCR) is the conversion of scanned images of printed, handwritten or typewritten text into machine-encoded text. This technology allows to automatically recognizing characters through an optical mechanism. In case of human beings, our eyes are optical mechanism. The image seen by eyes is input for brain. OCR is a technology that functions like human ability of reading. OCR is not able to compete with human reading capabilities. OCR is a technology that enables you to convert different types of documents such as scanned paper documents, PDF files or images captured by a digital camera into editable and searchable data.

One widely known application is in banking, where OCR is used to process demand draft or cheque without human involvement. An image of demand draft or cheque can be captured by mobile camera, the writing on it is scanned instantly, and the correct amount of money is transferred. This technology has nearly been perfected for printed demand draft or cheque, and is fairly accurate for handwritten demand draft or cheque as well, though it requires signature verification. In the legal industry, there has also been a significant movement to digitize paper documents. In order to save space and eliminate the need to sift through boxes of paper files, documents are being scanned. OCR further simplifies the process by making documents text-searchable, so that they are easier to locate and work with once in the database. Legal professionals now have fast, easy access to a huge library of documents in electronic format, which they can find simply by typing in a few keywords. OCR is widely used in many other fields, including education, finance, and government agencies.

In this paper, one effective optical character recognition from text image using texture and topological features is proposed. For better performance, the texture and topological features of all characters of text image like corner points, features of different regions, and ratio of character area and convex area are calculated. Based on the texture and topological information, character verification is done using feature matching between the extracted character and the template of all character serves as a measure of similarity between the two. This paper is organized into the following sections. Section II describes an overview of previous work. Implementation details for optical character recognition are mentioned in section III. Experimented results are shown in section IV. Finally, the conclusions are in section V.

2. Related Work

Various methods have been proposed in the past for detection and localization of text in images and videos. These approaches take into consideration different properties related to text in an image such as color, intensity, connected-components, edges etc. These properties are used to distinguish text regions from their background and/or other regions within the image. The algorithm proposed by Wang and Kangas in [1] is based on color clustering. The input image is first pre-processed to remove any noise if present. Then the image is grouped into different color layers and a gray component. This approach utilizes the fact that usually the color data in text characters is different from the color data in the background. The potential text regions are localized using connected component based heuristics from these layers. Also an aligning and merging analysis (AMA) method is used in which each row and column value is analyzed [1]. The experiments conducted show that the algorithm is robust in locating mostly Chinese and English characters in images; some false alarms occurred due to uneven lighting or reflection conditions in the test images.

The text detection algorithm in [2] is also based on color continuity. In addition it also uses multi-resolution wavelet transforms and combines low as well as high level image features for text region extraction. The text finder algorithm proposed in [3] is based on the frequency, orientation and spacing of text within an image. Texture based segmentation is used to distinguish text from its background. Further a bottom-up ‘chip generation’ process is carried out which uses the spatial cohesion property of text characters. The

chips are collections of pixels in the image consisting of potential text strokes and edges. The results show that the algorithm is robust in most cases, except for very small text characters that are not properly detected. Also in the case of low contrast in the image, misclassifications occur in the texture segmentation.

A focus of attention based system for text region localization has been proposed by Liu and Samarabandu in [4]. The intensity profiles and spatial variance is used to detect text regions in images. A Gaussian pyramid is created with the original image at different resolutions or scales. The text regions are detected in the highest resolution image and then in each successive lower resolution image in the pyramid.

The approach used in [5, 6] utilizes a support vector machine (SVM) classifier to segment text from non-text in an image or video frame. Initially text is detected in multi-scale images using edge based techniques, morphological operations and projection profiles of the image [6]. These detected text regions are then verified using wavelet features and SVM. The algorithm is robust with respect to variance in color and size of font as well as language.

3. Proposed Methodology

Optical character recognition (OCR) takes a text image as input and gives editable text document as output. The OCR system primarily involves four steps: Pre-processing, Features extraction, Features training, and Feature matching. Flow chart of the OCR is shown in Figure 1. Here, two data sets are considered, one for training dataset and another for test dataset. Preprocessing and feature extraction is done in both cases. Features extracted from test data is compared with features extracted from training data to get the desired output.

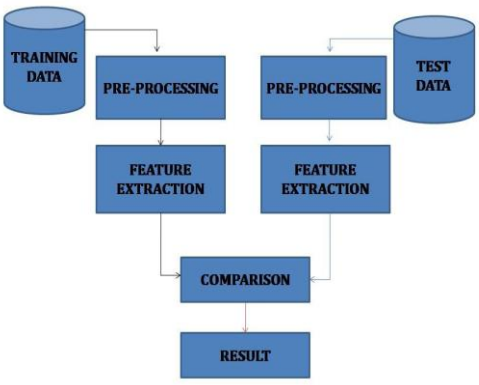

Figure 1: Flowchart of OCR system

3.1 Pre-Processing

The text image is converted into binary image for further working as shown in Figure 2 and Figure 3. When any image is converted to binary, it is easy to work with pixel values 0 and 1. The binary image is complimented so that the letters constitute by binary 1 (one) and background constitute by binary 0 (zero) as shown in Figure 4.

Figure 2: Text Image Figure 3: Binary Image

Now, individual text lines are separated from the binary image. This is done by calculating the sum of all values in a row. When the sum is 0, a new line is identified and separation is done. The sum of all rows in between two lines should be zero. The image is divided into several lines and each line is extracted one by one as shown in Figure 5. This procedure is repeated until all lines are extracted.

Figure 4: Complimented Binary Image

Figure 5: Extracted lines

Single lines are extracted due to the fact that, dealing with one line is easier than dealing with the whole image. Again, for each line, the letters are to be extracted as shown in Figure 6 and Figure 7. This is done by calculating the sum of all values in a column. When sum is zero, a character is identified and separation is done. In this way, all individual characters (alphabets, digits, punctuations) are separated.

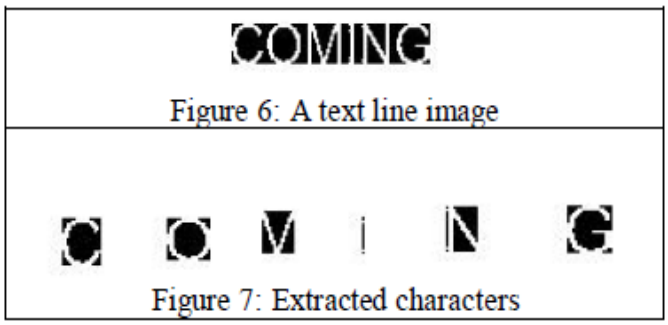

Figure 6: A text line image

Figure 7: Extracted characters

3.2 Features Extraction

Feature extraction technique is applied for all individual extracted characters. The character image is divided into four regions as shown in Figure 8.

Figure 8: Extracted character

Sum of the pixels value of the whole image and sum of pixels value in each of the sub-regions are calculated. Then their ratios are calculated as the features value of f1, f2, f3, f4 respectively.
f1= Sum of the pixels value of 1st quadrant /
Sum of the pixels value of the whole image
f2= Sum of the pixels value of 2nd quadrant /
Sum of the pixels value of the whole image
f3= Sum of the pixels value of 3rd quadrant /
Sum of the pixels value of the whole image
f4= Sum of the pixels value of 4th quadrant /
Sum of the pixels value of the whole image

To get better accuracy, features f5, f6, f7, f8, f9, and f10 are calculated using f1, f2, f3, and f4.
f5=f1+f2
f6=f2+f3
f7=f3+f4
f8=f1+f4
f9=f2+f4
f10=f1+f3

Using Harris corner method, numbers of corner points are calculated from character image. Feature f11 is considered as the number of corner points of a character. Total area of extracted character image is calculated using the actual number of pixels in the character image. Convex area of the character is calculated using the number of pixels in convex hull that can contain the character region. Feature f12 is ratio of convex area to total area.
f12= Convex Area / Total Area

Total twelve features f1 to f12 are extracted for all individual extracted characters.

3.3 Features Training

Here, three fonts, namely 'Lucida Fax', 'Berlin Sans' and 'Arial' have been considered as training data set. Three images Figure 9, Figure 10, and Figure 11 is used to extract the character features for training the system. The trained features value will be used for recognizing the extracted character.

Figure 9: Text Image of Lucida Fax

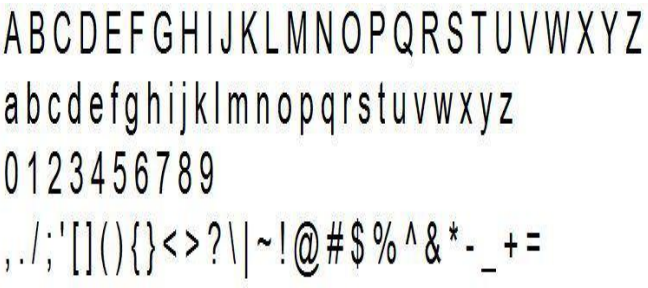

Figure 10: Text Image of Arial

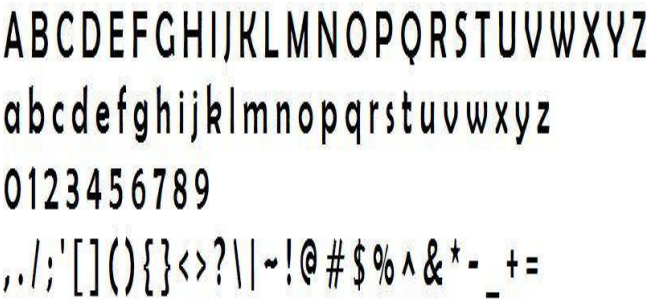

Figure 11: Text Image of Berlin Sans

3.4 Feature Matching

The features value is matched with the trained features set to recognize the exact character. Different matching algorithm can be used for feature matching. The minimum distance value with respect to all the features (f1 to f12) is selected as required character.

Algorithm for OCR
STEP 1: The input text image is converted into binary image.
STEP 2: The binary image is complimented so that the letters constitute by binary 1 (one) and background constitute by binary 0 (zero).
STEP 3: All text lines are separated from the binary image. This is done by finding the sum of all values in a row. When the sum is 0, a new line is identified and separation is done. The sum of all rows in between two lines should be zero.
STEP 4: For each line, the characters are to be extracted. This is done by finding the sum of all pixels value in a column. When sum is zero, a new character is identified and separation is done.
STEP 5: Total 12 features value f1 to f12 are extracted for each character.
STEP 6: The features value are matched with the trained features set to recognize the exact character.

4. Implementation

4.1Binarization

1) Binarization is the first step in the pre-processing of a document analysis and recognition system.
2) Binarizing the text bounded by text regions and marking text as one binary level and background as the other.
3) In binarization the mid value of RGB is taken i.e. 128
4) Hence if the RGB value is above 128 then that part will be converted into white & if

The RGB value is below 128 that part will be converted into black

Pseudo Code for Binarization

for all pixels(X,Y) in a CC
if intensity(X,Y) < (Gmin + Gmax)/2,then
mark(X,Y) as foreground
else
if no. of foreground neighbors > 4,then
mark(X,Y) as foreground
else
mark(X,Y) as background
end if
end if
end for

Figure 12: Binarization

4.2 Median Filter

1) The median filter is a nonlinear digital filtering technique, often used to remove noise.
2) Such noise reduction is a typical pre-processing step to improve the results of later processing
3) Thus median filter makes the image smooth and delete all the spots and noise.
4) Median filter will help to improve the efficiency of the output
5) Median filter will make the image spots remove

Figure 13: Median Filter

4.3 Segmentation

1. Image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.
2. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

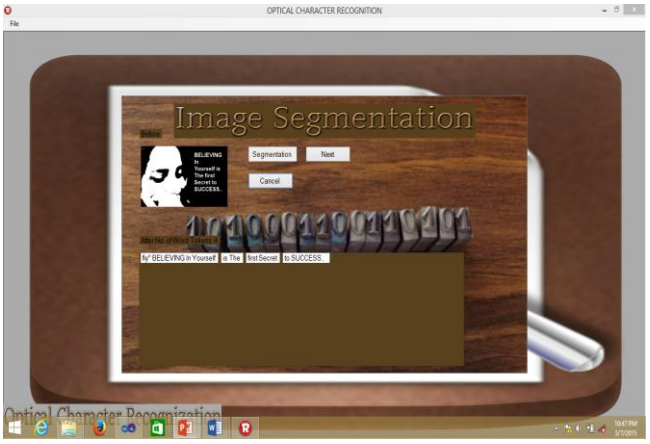

Figure 14: Segmentation

4.4 Text Detection and Extraction

1. Text detection refers to the determination of the presence of text in a given frame.
2. Text extraction is the stage where the text components are segmented from the background.
3. The extracted text images can be transformed into plain text using OCR technology.

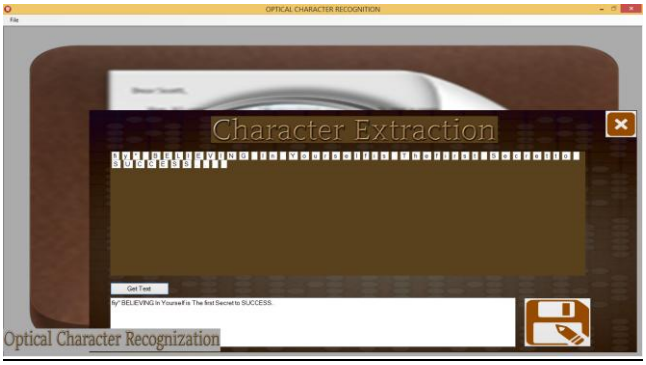

Figure 15: Character Extraction

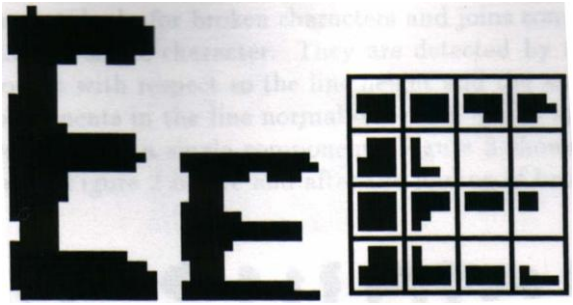

Figure 16: Example of Character Detection

5. Result and Discussion

OCR is generally an "offline" process, which analyzes a static document. Handwriting movement analysis can be used as input to handwriting recognition. Instead of merely using the shapes of glyphs and words, this technique is able to capture motions, such as the order in which segments are drawn, the direction, and the pattern of putting the pen down and lifting it. This additional information can make the end-to-end process more accurate. This technology is also known as "on-line character recognition", "dynamic character recognition", "real-time character recognition", and "intelligent character recognition".

Commissioned by the U.S. Department of Energy (DOE), the Information Science Research Institute (ISRI) had the mission to foster the improvement of automated technologies for understanding machine printed documents, and it conducted the most authoritative of the Annual Test of OCR Accuracy from 1992 to 1996.

Recognition of Latin-script, typewritten text is still not 100% accurate even where clear imaging is available. One study based on recognition of 19th- and early 20th-century newspaper pages concluded that character-by-character OCR accuracy for commercial OCR software varied from 81% to 99%; total accuracy can be achieved by human review or Data Dictionary Authentication. Other areas—including recognition of hand printing, cursive handwriting, and printed text in other scripts (especially those East Asian language characters which have many strokes for a single character)—are still the subject of active research. The MNIST database is commonly used for testing systems' ability to recognize handwritten digits.

Accuracy rates can be measured in several ways, and how they are measured can greatly affect the reported accuracy rate. For example, if word context (basically a lexicon of words) is not used to correct software finding non-existent words, a character error rate of 1% (99% accuracy) may result in an error rate of 5% (95% accuracy) or worse if the measurement is based on whether each whole word was recognized with no incorrect letters.

Web based OCR systems for recognizing hand-printed text on the fly have become well known as commercial products in recent years Accuracy rates of 80% to 90% on neat, clean hand-printed characters can be achieved by pen computing software, but that accuracy rate still translates to dozens of errors per page, making the technology useful only in very limited applications.

Recognition of cursive text is an active area of research, with recognition rates even lower than that of hand-printed text. Higher rates of recognition of general cursive script will likely not be possible without the use of contextual or grammatical information. For example, recognizing entire words from a dictionary is easier than trying to parse individual characters from script. Reading the Amount line of a cheque (which is always a written-out number) is an example where using a smaller dictionary can increase recognition rates greatly. The shapes of individual cursive characters themselves simply do not contain enough information to accurately (greater than 98%) 5 recognize all handwritten cursive script.

6. Conclusion

A number of methods have been proposed by several authors for optical character recognition. A new method to extract features from text images and recognition of exact character to produce text document is presented here. The proposed method promises a very simple but reliable solution to the problem of optical character recognition. The technique that is used based on calculating the number of corner points and utilizing the various properties like object area and convex areas of the image. The proposed algorithm will help the community in the field of character recognition. By introducing more features, the accuracy can be enhanced.

References

[1] Kongqiao Wang and Jari A. Kangas, Character location in scene images from digital camera, The journal of the Pattern Recognition society, March 2003.
[2] K.C. Kim, H.R. Byun, Y.J. Song, Y.W. Choi, S.Y. Chi, K.K. Kim and Y.K Chung, Scene Text Extraction in Natural Scene Images using Hierarchical Feature Combining and verification, Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04), IEEE.
[3] Victor Wu, Raghavan Manmatha, and Edward M. Riseman, TextFinder: An Automatic System to Detect and Recognize Text in Images, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 11, November 1999.
[4] Xiaoqing Liu and Jagath Samarabandu, A Simple and Fast Text Localization Algorithm for Indoor Mobile Robot Navigation, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 5672, 2005.
[5] Qixiang Ye, Qingming Huang, Wen Gao and Debin Zhao, Fast and Robust text detection in images and video frames, Image and Vision Computing 23, 2005.
[6] Qixiang Ye, Wen Gao, Weiqiang Wang and Wei Zeng, A Robust Text Detection Algorithm in Images and Video Frames, IEEE, 2003.
[7] Victor Wu, Raghavan Manmatha, and Edward M. Riseman, TextFinder: An Automatic System to Detect and Recognize Text in Images, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 11, November 1999.
[8] Xiaoqing Liu and Jagath Samarabandu, A Simple and Fast Text Localization Algorithm for Indoor Mobile Robot Navigation, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 5672, 2005.
[9] Qixiang Ye, Qingming Huang, Wen Gao and Debin Zhao, Fast and Robust text detection in images and video frames, Image and Vision Computing 23, 2005.
[10] Rainer Lienhart and Axel Wernicke, Localizing and Segmenting Text in Images and Videos, IEEE Transactions on Circuits and Systems for Video Technology, Vol.12, No.4, April 2002.
[11] Qixiang Ye, Wen Gao, Weiqiang Wang and Wei Zeng, A Robust Text Detection Algorithm in Images and Video Frames, IEEE, 2003.

[12] Debapratim Sarkar, Raghunath Ghosh ,A bottom-up approach of line segmentation from handwritten text ,2009.
[13] Karin Sobottka, Horst Bunke and Heino Kronenberg, Identification of text on colored book and journal covers, Document Analysis and Recognition, 20-22, 1999.
[14] Yingzi Du Chein-I Change and Paul D.Thouin, Automated system for text detection in individual video images, Journal of Electronic Imaging ,pp410-422, July 2003.
[15] W.Mao, F. Chung, K. Lanm, and W.Siu, Hybrid chinese/english text detection in images and videos frames, Proc.of International Conference on Pattern Recognition,2002,vol.3, pp. 1015-1018
[16] Bassem Bouaziz, Walid Mahdi, Mohsen Ardabilain, Abdelmajid Ben Hamadou, A New Approach For Texture Features Extraction: Application For Text Localization In Video Images IEEE,ICME 2006.
[17] Bassem Bouaziz, Walid Mahdi, Mohsen Ardabilain, Abdelmajid Ben Hamadou, A New Approach For Texture Features Extraction: Application For Text Localization In Video Images IEEE,ICME 2006.
[18] C. Wolf, J. –M. Jolion F. Chassaing , Text localization, enhancement and binarization in multimedia documents , Patterns Recognition ,2002.proceedings. 16th International Conference on ,vol 2,11-15, pp.1037-1040, Aug. 2002.
[19] Jui-Chen Wu · Jun-Wei Hsieh Yung-Sheng Chen, Morphology-based text line extraction, Machine